Command, Control, and Navigation
Experimental Results With the NPS ARIES AUV

David B. Marco
Dept. of Mechanical Engineering
Naval Postgraduate School
Monterey, CA

Anthony J. Healey
Dept. of Mechanical Engineering
Naval Postgraduate School
Monterey, CA

Abstract-**This paper provides an overview of the Naval Postgraduate School ARIES autonomous underwater vehicle and its control and navigation performance. An attempt is made to highlight its current operational capabilities and provide a description of future enhancements for greater mission utility and flexibility. An overview of the vehicle design along with descriptions of all major hardware components and sensors is given. A major discussion of the implementation of a modular, multi-rate, multi-process software architecture for the ARIES is provided. The architecture is designed to operate using either a single computer processor or two independent, cooperating processors linked through a network interface for improved load balancing. A dual computer implementation is presented here since each processor assumes different tasks for mission operation. Also included is a section on the underwater navigation method using a real-time extended Kalman filter that fuses all sensor data and computes the real time position, orientation, velocity, etc., of the vehicle. Experimental results for navigational accuracy using a DGPS/IMU/ Doppler aided navigation system are presented with DGPS pop-up maneuvers. Navigational accuracy is a requirement for the use of ARIES as a mobile communications network node. This work provides some examples of missions possible with such a node and the current state of its command and control system.**

I. INTRODUCTION

The use of AUVs for Ocean Survey and military mine countermeasures is well documented, [1],[2], and [3] among many others. The continuing problems facing further commercial use of such systems include accurate underwater navigation and communications links. Further research in low cost navigational accuracy and communications links is needed, and toward that end, this paper presents a description of the latest generation of the Naval Postgraduate School (NPS) underwater vehicle named the ARIES AUV. The ARIES vehicle is a shallow water communications server vehicle with a Differential Global Positioning System (DGPS) and a doppler aided Inertial Measurement Unit (IMU) / Compass navigation suite. Navigational errors are corrected by DGPS when surfaced, which, for shallow water applications presents no penalty. The vehicle is shown in a DGPS pop-up maneuver in Monterey Bay in Figure 1, and Figure 2 shows the command and control system as it currently exists.

The hull was outfitted in the fall of 1999 and has recently become fully operational (Spring 2000). The vehicle has been designed to test and demonstrate technology for a network server platform and target reacquisition missions, and was initially operated during AUVFest '99 in Gulfport, MS [4].

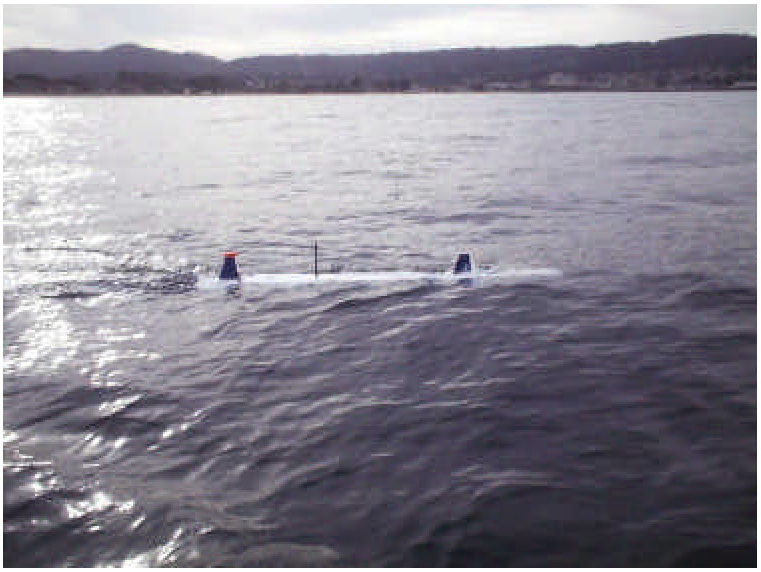

Figure 1. The NPS ARIES AUV in a GPS pop-up maneuver (December 2000).

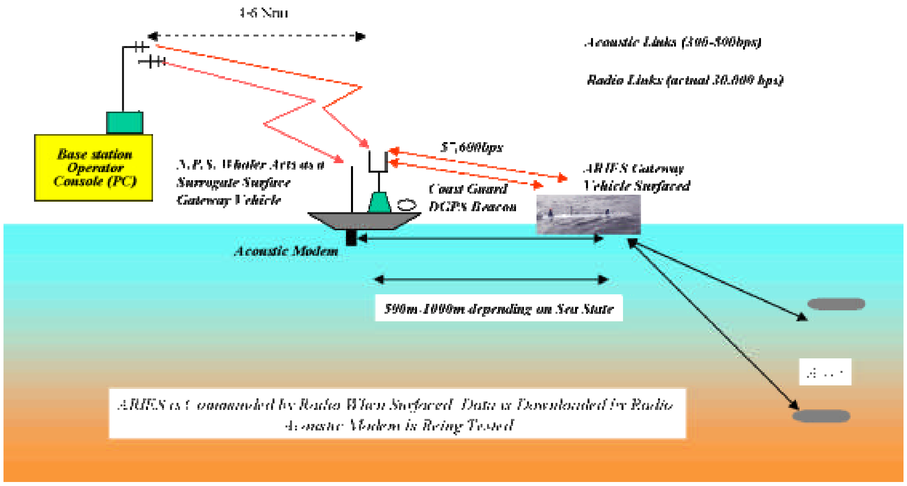
4-6 Nm
Acoustic Links (300-500bps)
Radio Links (actual 30.000 bps)
Base station Operator Console (PC)
N.P.S. Whaler Acts as a Surrogate Surface Gateway Vehicle
57,600bps
ARIES Gateway Vehicle Surfaced
Coast Guard DGPS Beacon
Acoustic Modem
500m-1000m depending on Sea State
ARIES is Commanded by Radio When Surfaced. Data is Downloaded by Radio
Acoustic Modem is Being Tested

Figure 2.ARIES current command and control system, showing its use as a possible gateway vehicle.

II. VEHICLE DESCRIPTION

The major hardware components of the ARIES are shown in Figure 3, and are described below.

Dimensions and Endurance: The vehicle weighs 225 Kg and measures approximately 3 m long, 0.4 m wide and 0.25 m high. The hull is constructed of ¼" thick 6061 aluminum and forms the main pressure vessel that houses all electronics, computers, and batteries. A flooded fiberglass nose is used to house the external sensors and power on/off switches and status indicators. It is capable of a top speed of 3.5 knots and is powered by six 12 volt rechargeable lead acid batteries. The endurance is approximately 3 hours at top speed, 20 hours hotel load only. The ARIES was primarily designed for shallow water operations and can operate safely down to 30 meters. However, finite element analysis has shown that a depth of 100 meters could be attained if the hull was strengthened in the central hatch area. Principally, using horizontal and vertical T-stiffeners, stresses and deflections of the central box could be reduced to acceptable levels with only a small loss of internal volume and added weight. The design using a box section is unique and is advantageous for component packaging with a small sacrifice of its hydrodynamic properties.

Propulsion and Motion Control Systems: Main propulsion is achieved using twin ½Hp electric drive thrusters located at the stern. During normal flight, heading and depth is controlled using upper bow and stern rudders and a set of bow planes and stern planes. Since the control fins are ineffective during very slow or zero forward speed maneuvers, vertical and lateral cross-body thrusters will be used to control surge, sway, heave, pitch, and yaw motions. While not installed at the time on ARIES, cross-body thruster control has been developed and presented for the Phoenix AUV [5].

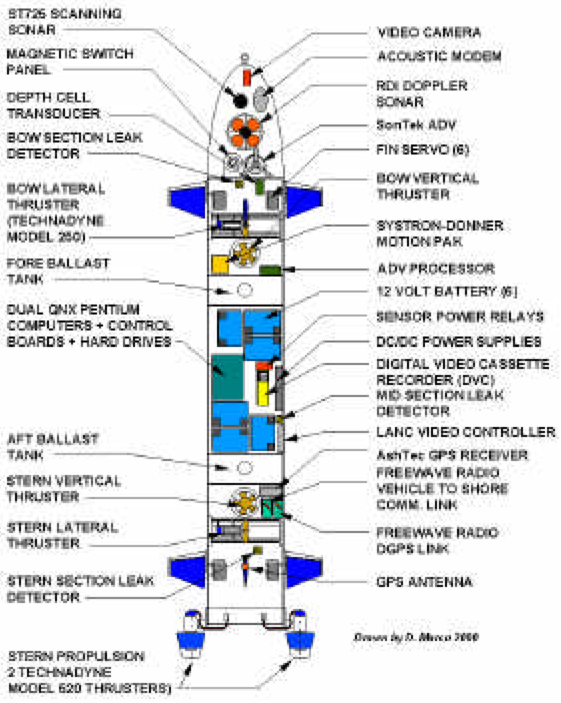

Figure 3. Hardware Components of the NPS ARIES

Navigation Sensors: The sensor suite used for navigation includes a 1200 kHz RD Instruments Navigator Doppler Velocity Log (DVL) that also contains a TCM2 magnetic compass. This instrument measures the vehicle ground speed, altitude, and magnetic heading. Angular rates and accelerations are measured using a Systron Donner 3-axis Motion Pak IMU that is considered to be a low cost "tactical" grade IMU. While surfaced, differential GPS (Ashtech G12-Sensor [6]), accuracy 40 cm, is available to correct any navigational errors accumulated during the submerged phases of a mission. In addition, and because of inaccuracies in the TCM2 compass, a Honeywell HMR3000 magneto-restrictive compass, corrected by a deviation table, is used as the primary heading reference standard. Experiments have shown that the deviation table maximum error is approximately 4 degrees in some orientations.

Sonar and Video Sensors: A Tritech ST725 scanning sonar or an ST1000 profiling sonar is used for obstacle avoidance and target acquisition/reacquisition. The sonar heads can scan continuously through 360^o of rotation or be swept through a defined angular sector. A fixed focus wide-angle video camera (Deep Sea Power and Light - SS100) is located in the nose and connected to a Digital Video Cassette (DVC) recorder. The computer is interfaced to the recorder and controls on/off and start/stop record functions. While recording, the date, time, vehicle position, depth and altitude is superimposed on the video image.

Vehicle/Operator Communications: Freewave Radio Modems are used for moderate bandwidth (2000-3000 bytes/sec over 4 to 6 nautical miles with repeaters) command and control (C^2), between command center and the vehicle when surfaced as shown in Figure 2. Kermit file transfer protocol is used in the vehicle computer with Zmodem through Procomm protocol on the base station side.

In recent experiments we have routinely transferred data files between the surfaced ARIES, a Boston Whaler repeater station, and a base station command center. Radio modem connections require line-of-sight and are critically dependent on antennae height above ground. In this case, the ARIES antenna reaches to about 45 cm above the water-line. The repeater aerials on the Boston Whaler are about 5 m above the water-line. With 2-3 m swells in Monterey Bay, range is limited to 600 m before the waves cause intermittent communications. The range between the base station antenna at 30 m above the water (pier) and the Whaler repeater link (5 m) can be as large as 6 km on a calm day without degradation of the C^2 link.

The differential RTCM corrections are transmitted from the U. S. Coast Guard and sensed using an ABX beacon receiver on the Whaler. Both C^2 data and the RTCM data are sent as two separate channels to the single antenna on ARIES. Internally, ARIES splits the data channels and distributes them respectively to the G12 sensor and the control computer.

Necessary for its use as a network server or gateway, is the ability of ARIES to communicate underwater with other vehicles. At the time of writing, ARIES carries an FAU acoustic modem, details of which are provided in [7]. The modem was easily installed in the ARIES since the transducer head was placed on the vehicle nose connected through the forward bulkhead to the electronic boards in the dry compartment. Other modems could be installed in the same fashion allowing for more than one modem to be used during the same mission. This would allow future networking links between different organization’s vehicles without an interoperable standard in place.

What has been shown to date is that over a 150 m range in Monterey harbor, error free communications to ARIES with specified replies, have been demonstrated with down to 20% of full transmission power with all other systems active.

Results for the integration of both acoustic and radio links through ARIES are yet to be completed, but will be the subject of a future paper.

III. COMPUTER HARDWARE ARCHITECTURE

The dual computer system unit consists of two Ampro Little Board 166 MHz Pentium computers with 64 MB RAM, four serial ports, a network adapter, and a 2.5 GB hard drive each. Two DC/DC voltage converters for powering both computer systems and peripherals are integrated into the computer package. The entire computer system draws a nominal 48 Watts. Both systems use TCP/IP sockets over thinwire ethernet for inter processor communications and connections to an external LAN. The sensor data gathering computer is designated QNXT, while the second is named QNXE and executes the various auto-pilots for servo level control. Both computers are used as the baseboard for a stack of Diamond Systems PC-104 data acquisition boards.

IV. COMPUTER SOFTWARE ARCHITECTURE

A. The Architecture

A diagram outlining the modular, multi-rate, multi-process software architecture is shown in Figure 4. The architecture is designed to operate using a single computer processor or two, independent, co-operating processors linked through a network interface. Splitting the processing between two computers can significantly improve computational load balancing and software segregation. A dual computer implementation is presented here, since, in the ARIES, each processor assumes different tasks for mission operation. Both computers run the QNX real time operating system using synchronous socket sender and receiver network processes for data sharing between the two. On each processor, inter-process communication is achieved using semaphore controlled shared memory structures. Deadlocks and race conditions are explicitly eliminated by the careful use of semaphores in this system design. At boot time, the network processes are started automatically and all shared memory segments are created in order to minimize the amount of manual setup performed by the user**.**

All vehicle sensors are interrogated by separate, independently controlled, ***concurrent*** processes, and there is no restriction on whether the processes operate ***synchronously*** or ***asynchronously***. Since various sensors gather data at different rates, each process may be tailored to operate at the acquisition speed of the respective sensor. Each process may be started, stopped, or reset independently allowing easy reconfiguration of the sensor suite needed for a given mission. All processes are written in C.

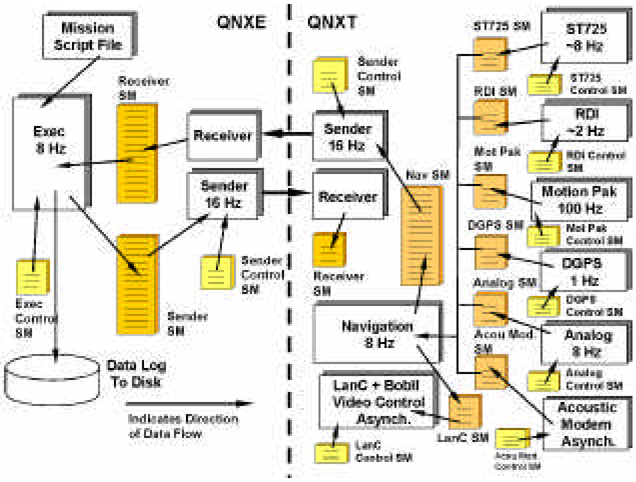

Figure 4. Dual computer software architecture. Blocks denoted "SM" are shared memory segments while all others refer to concurrent processes.

To allow synchronous sensor fusion, each process contains a unique shared memory data structure that is updated at the specific rate of each sensor. All sensor data are accessible to a ***synchronous*** navigation process through shared memory and is a main feature of the software architecture. Incorporated into the navigation process is an extended Kalman filter that fuses all sensor data and computes the real time position, orientation, velocity, etc, of the vehicle. The dual computer implementation uses one processor for data gathering and running the navigation filters, while the second uses the output from the filters to operate the various auto-pilots for servo level control. Once the state information is computed, it is transmitted to the second computer over standard TCP/IP sockets.

B. Mission Control Modes

All vehicle behaviors are determined by a pre-programmed mission script file. This is parsed in the QNXE computer by the process *Exec*. The file contains a sequential list of commands that the vehicle is to follow during a mission. These commands may be as simple as setting the stern propulsion thruster speeds, to more complex maneuvers such as commanding the vehicle to repeatedly fly over a submerged target at a given GPS coordinate using altitude and cross-track error control.

Below is the mission script file used for the results section of this paper.

```
USE_CTE_WAYPOINT_CONTROL Track.out
USE_TIME_BASED_CONTROL
#                            L rpm R rpm
SET_SCREW_SPEED              500.0 500.0
USE_SCREW_SPEED_CONTROL
#                            Depth (m)
SET_DEPTH                    0.0
#                            RudderAngle (Deg)
SET_FIXED_RUDDER_ANGLE       0.0
USE_FIXED_RUDDER_CONTROL
USE_FLIGHT_DEPTH_CONTROL
#                            Time (Sec)
SET_FLIGHT_DURATION          10.0
SET_SCREW_SPEED              0.0  0.0
#                            Time (Sec)
SET_FLIGHT_DURATION          5.0
SHUTDOWN
```

The main function of the mission script above is to use cross-track error control given by the keyword `USE_CTE_WAYPOINT_CONTROL` with a predefined set of way points in the file *`Track.out`*, given in Appendix A. Once the pattern has been completed, time based control is used in which all phases are performed for a specified time. In this case, the vehicle is commanded to set the twin stern thruster speeds at 500 rpm, fix the rudder at 0 degrees, and set the depth to 0 meters for a duration of 10 seconds. This is usually a sufficient amount of time to surface the vehicle. After the 10 second surfacing maneuver, the thruster speeds are set to 0 rpm for 5 seconds, and finally the mission terminates using the keyword `SHUTDOWN`.

V. ARIES AUTOPILOT CONTROL LAWS

The NPS ARIES currently uses four different auto pilots for flight maneuvering control. They consist of independent diving, steering, altitude above bottom, and cross-track error controllers. All four auto pilots are based on sliding mode control theory and each mode (i.e. diving, steering) is de-coupled for ease of implementation and design. A reference for the details of controller design methodology may be found in [8]. These control laws are not unique since fuzzy and heuristic control is used in the Florida Atlantic University (FAU) vehicles [9]. However, the authors here have found that Sliding Mode controllers are simple to use and implement with minimal tuning.

A. Depth Controller

Since the vehicle depth can be independently controlled by the dive planes alone, the diving controller may be modeled by a linearized system with a single generalized input control, *u(t)*, generating a pitch-dive control distributed to bow and stern planes in an equal and opposite amount, and is of the form

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{b}\,u(t)\,, \tag{1}$$

and for the ARIES, the dynamics are given by the system of equations

$$\begin{bmatrix}\dot{q}(t)\\ \dot{q}(t)\\ \dot{Z}(t)\end{bmatrix} = \begin{bmatrix}-1.3899 & -0.0032 & 0\\ 1 & 0 & 0\\ 0 & -U & 0\end{bmatrix}\begin{bmatrix}q(t)\\ q(t)\\ Z(t)\end{bmatrix} + \begin{bmatrix}-2.6091\\ 0\\ 0\end{bmatrix} d_{sp}(t) + (disturbances) \tag{2}$$

where $q(t)$ is the pitch rate, $q(t)$ is the pitch angle, $Z(t)$ is the depth in meters, and $d_{sp}(t)$ is the stern plane angle in radians. U is the nominal longitudinal speed of the vehicle expressed in (m/sec) and a value of 1.8 m/sec is used. Although the bow and stern planes may be independently controlled, currently both sets of planes operate as coupled pairs such that the command to the bow planes is $-d_{sp}(t)$. Notice that the heave velocity, $w(t)$, equation is ignored, as also are its effects on the $q(t)$ and $Z(t)$ equations of motion. They are considered to be disturbances. The reduction of the system to third order creates a simplification that is both valid and useful.

The sliding surface is then formed as a linear combination of state variable errors in the usual way. Ignoring any nonzero pitch angle and rate commands, the sliding surface polynomial using a commanded depth below the surface, Z_{com}, becomes

$$\acute{o}(t) = 0.7693q(t) + 0.6385\grave{e}(t) + 0.0724(Z_{com} - Z(t)) \tag{3}$$

and the corresponding control law for the stern planes is

$$\ddot{a}_{sp}(t) = 0.4994(-0.4105q(t) + 0.1086\grave{e}(t) + \c{c}\, tanh(s\,(t)\,/\,f\,)) \tag{4}$$

where $h = 1.0$ and $f = 0.5$.

B. Altitude Controller

In order to control the vehicle altitude above the bottom designated $h(t)$, we simply need to change some of the signs of the terms from the diving equations. Noting the sign difference of the pitch angle and rate coefficients, this results in the following sliding surface

$$\acute{o}(t) = -0.7693q(t) - 0.6385\grave{e}(t) + 0.0724(h_{com} - h(t)) \ . \tag{5}$$

The stern plane command for altitude control is

$$d_{sp}(t) = -\,0.4994(\,0.4105q(t) - 0.1086q(t) + h\, t\,anh(s\,(t)/f\,)) \tag{6}$$

where $h = 1.0$, $f = 0.5$, and h_{com} is the commanded altitude above bottom.

C. Heading Controller

By similar reasoning, and to eliminate the need to feedback the side-slip velocity, we argue that a second order model is sufficient for heading control. The side-slip effects are treated as disturbances that the control overcomes. Thus the heading model becomes

$$\begin{aligned} \dot{r}(t) &= ar(t) + bd_r(t) + (disturbances) \\ \dot{y}(t) &= r(t) \end{aligned} \tag{7}$$

where $y(t)$ is the vehicle heading angle, $r(t)$ is the yaw rate, and $d_r(t)$ is the stern rudder angle. The coefficients a and b have been determined using system identification techniques from past in-water experiments and are $a = -0.30\,sec^{-1}$ and $b = -0.1125\ sec^{-2}$. The stern and bow rudders operate in the same way as the planes, therefore, the command to the bow rudder is $-d_r(t)$.

Notice that in order to use this steering law, the heading error $(y_{com} - y(t))$ must lie between $\pm 180^0$, and is de-wrapped as needed in order to make that happen. Ignoring any nonzero commanded yaw rate, the sliding surface is defined by

$$s(t) = -0.9499\, r(t) + 0.1701(y_{com} - y(t)). \tag{8}$$

The stern rudder command for heading control is

$$d_r(t) = -1.543(2.5394r(t)\ +\ ç\, tanh(s(t)/f)) \tag{9}$$

where $h = 1.0$ and $f = 0.5$.

D. Cross-Track Error Controller

To follow a set of straight line tracks that form the basis of many guidance requirements, a sliding mode controller is presented that has been experimentally validated under a wide variety of conditions. Other works have studied this problem for land robots, for example, [10], and usually develop a stable guidance law based on cross-track error. Here, with Figure 5 as a guide to the definitions, we use a combination of a line of sight guidance (LOS) [8] and a cross-track error control (CTE). With large heading errors, the cross-track error control can not be guaranteed stable, while a line of sight heading control will reduce heading errors to zero. Switching between these two controllers allows for reduction of both cross-track and heading errors under all conditions.

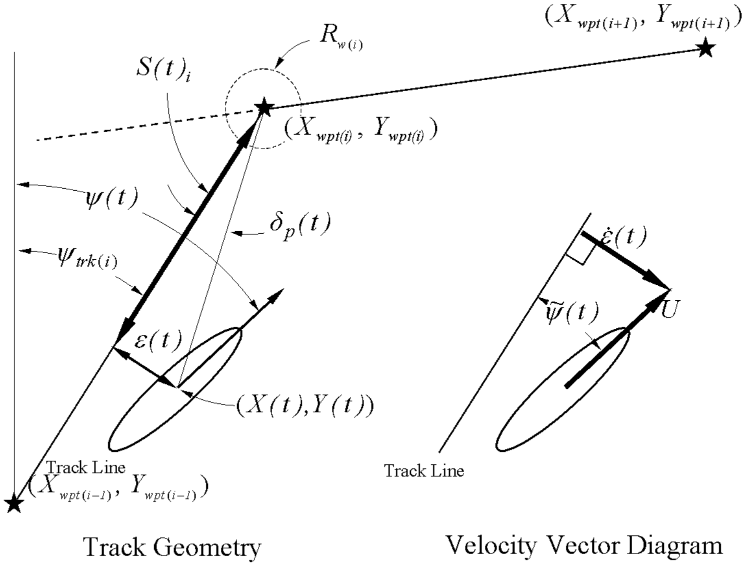

Figure 5. Track geometry and velocity vector diagram.

The variable of interest to minimize is the cross-track error, $e(t)$, and is defined as the perpendicular distance between the center of the vehicle (located at $(X(t), Y(t))$) and the adjacent track line. The total track length between way point *i* and *i-1* is given by

$$L_i = \sqrt{(X_{wpt(i)} - X_{wpt(i-1)})^2 + (Y_{wpt(i)} - Y_{wpt(i-1)})^2}\ , \tag{10}$$

where the ordered pairs $(X_{wpt(i)}, Y_{wpt(i)})$ and $(X_{wpt(i-1)}, Y_{wpt(i-1)})$ are the current and previous way points respectively. The track angle, $y_{trk(i)}$, is defined by

$$y_{trk(i)} = arctan2(Y_{wpt(i)} - Y_{wpt(i-1)}, X_{wpt(i)} - X_{wpt(i-1)})\ . \tag{11}$$

and is a constant for a given set of adjacent way points. The cross-track heading error $\tilde{y}(t)_{CTE(i)}$ for the i^{th} segment is defined as

$$\tilde{y}(t)_{CTE(i)} = y(t) - y_{trk(i)} \tag{12}$$

where $\tilde{y}(t)_{CTE(i)}$ must be normalized to lie between $\pm 180^0$. The difference between the current vehicle position and the next way point is

$$\begin{aligned} \tilde{X}(t)_{wpt(i)} &= X_{wpt(i)} - X(t) \\ \tilde{Y}(t)_{wpt(i)} &= Y_{wpt(i)} - Y(t) \end{aligned} \tag{13}$$

With the above definitions, the distance to the i^{th} way point projected to the track line $S(t)_i$ can be calculated using

$$S(t)_i = [\tilde{X}(t)_{wpt(i)} \;\; \tilde{Y}(t)_{wpt(i)}] \cdot [(X_{wpt(i)} - X_{wpt(i-1)}) \;\; (Y_{wpt(i)} - Y_{wpt(i-1)})]/L_i \tag{14}$$

therefore, $S(t)_i$ ranges from 0-100% of L_i.

The cross-track error $e(t)$ may now be defined as

$$e(t) = S(t)_i \, sin(\ddot{a}_p(t)) \tag{15}$$

where $\ddot{a}_p(t)$ is the angle between the line of sight to the next way point and the current track line given by

$$\begin{aligned} \ddot{a}_p(t) = \; & arctan\,2(Y_{wpt(i)} - Y_{wpt(i-1)}, X_{wpt(i)} - X_{wpt(i-1)}) \\ & - arctan\,2(\tilde{Y}(t)_{wpt(i)}, \tilde{X}(t)_{wpt(i)}) \end{aligned} \tag{16}$$

and must be normalized to lie between $\pm 180^0$, and *arctan2* is the inverse tangent function atan2, as defined in Matlab.

With the cross-track error defined, the sliding surface can be cast in terms of derivatives of the errors such that

$$\begin{aligned}
&\dot{e}(t) = U\,sin(\tilde{y}(t)_{CTE(i)}) \\
&\ddot{e}(t) = Ur(t)\,cos(\tilde{y}(t)_{CTE(i)}) \qquad\qquad\qquad\qquad 0 < \tilde{y}(t)_{CTE(i)} < p/2\,; \\
&\dddot{e}(t) = U\dot{r}(t)\,cos(\tilde{y}(t)_{CTE(i)}) - Ur(t)^2\,sin(\tilde{y}(t)_{CTE(i)})
\end{aligned}$$

recalling that U is the nominal longitudinal speed of the vehicle. The sliding surface for the cross-track error controller becomes a second order polynomial of the form

$$s\,(t) = \ddot{e}(t) + l_1\dot{e}(t) + l_2 e(t) \tag{17}$$

The condition for stability of the sliding mode controller is

$$\dot{s}\,(t) = \dddot{e}(t) + l_1\ddot{e}(t) + l_2\dot{e}(t) = -h(s\,(t)/f)\,, \tag{18}$$

and to recover the input for control, the heading dynamics, Equation (7), may be substituted into Equation (17) to obtain

$$\begin{aligned}
&U(ar(t) + bd_r(t))\,cos(\tilde{y}(t)_{CTE(i)}) - Ur(t)^2\,sin(\tilde{y}(t)_{CTE(i)}) + l_1Ur(t)\,cos(\tilde{y}(t)_{CTE(i)}) \\
&+ l_2U\,sin(\tilde{y}(t)_{CTE(i)}) = -h(s\,(t)/f) \quad ; \quad 0 < \tilde{y}(t)_{CTE(i)} < p/2
\end{aligned} \tag{19}$$

Rewriting Equation (17), the sliding surface becomes

$$s\,(t) = U\,r(t)\,cos(\tilde{\o}(t)_{CTE(i)}) + l_1U\,sin(\tilde{\o}(t)_{CTE(i)}) + l_2e(t)\,. \tag{20}$$

The rudder input can be expressed as

$$\begin{aligned} d_r(t) \;=\; & \left(\frac{1}{Ub\,cos(\tilde{\o}(t)_{CTE(i)})}\right)(-Uar(t)\,cos(\tilde{\o}(t)_{CTE(i)}) \\ & + U(r(t))^2\, sin(\tilde{\o}(t)_{CTE(i)}) - l_1 Ur(t)\,cos(\tilde{\o}(t)_{CTE(i)}) \\ & - l_2 U\, sin(\tilde{\o}(t)_{CTE(i)}) - ç(s\,(t)/f)) \quad ; \quad 0 < \tilde{y}(t)_{CTE(i)} < p/2 \end{aligned} \qquad (21)$$

where $l_1 = 0.6$, $l_2 = 0.1$, $h = 0.1$, and $f = 0.5$. To avoid division by zero, in the rare case where $cos(\tilde{\o}(t)_{CTE(i)}) = 0.0$ (i.e. the vehicle heading is perpendicular to the track line) the rudder command is set to zero since this condition is transient in nature. Note, if $\tilde{y}(t)_{CTE(i)} < p/2$, the vehicle will follow the track, but travel in the opposite direction to that desired. In order to prevent this from happening in practice, a bound of 40 degrees is used as a switch to LOS control.

Line of Sight Controller

When the condition arises that the magnitude of the cross-track heading error $\left|\tilde{\o}(t)_{CTE(i)}\right|$ exceeds 40 degrees, a LOS is used. In this situation, the heading command can be determined from

$$y\,(t)_{com(LOS)} = arc\,tan\,2(\tilde{Y}(t)_{wpt(i)}, \tilde{X}(t)_{wpt(i)}) \qquad (22)$$

and the LOS error from

$$\tilde{y}(t)_{LOS} = y\,(t)_{com(LOS)} - y\,(t)\,, \qquad (23)$$

and the control laws used for heading control, Equations (8, 9) may be used. The above condition is transient in nature and is normally required for the following cases. When the mission begins, the initial heading of the vehicle is seldom aligned with the commanded track. Line of sight control forces the vehicle to head in the direction of the current way point and once $\left|\tilde{\o}(t)_{CTE(i)}\right|$ reduces below 40 degrees, CTE control is used. The second case is when the angle between two sequential track lines exceed 40 degrees.

Two conditions may be true for the way point index to be incremented. The first and most usual case is if the vehicle has penetrated the way point watch radius $R_{w(i)}$. Secondly if a large amount of cross-track error is present, the next way point will become active if the projected distance to the way point $S(t)_i$ reaches some minimum value $S_{min(i)}$, such that

$$if \left(\sqrt{(\tilde{X}(t)_{wpt(i)})^2 + (\tilde{Y}(t)_{wpt(i)})^2} \;\pounds\; R_{w(i)} \;\; OR \;\; S(t)_i < S_{min(i)} \right) \quad \text{THEN}$$

$$Activate\ Next\ Way\ Point$$

In water experimental results using the controllers presented above will now be presented in the next section.

VI. NAVIGATION

The ARIES vehicle uses an INS / DOPPLER / DGPS navigational suite and an Extended Kalman Filter (EKF) which was developed and presented in [11], and may be tuned for optimal performance given a set of data. Because of space limits in this paper, the reader is referred to [11] for details of the filter, its states, and special data handling needs for working with multi-rate asynchronous sensors. The states include position, ground reference velocities, heading, yaw rate, and two error states, one each for the compass bias and the yaw rate bias. State estimates are automatically updated when sensor data is available. In particular, the error states are only corrected when GPS data becomes available. The filter heading state is initialized using the magnetic compass, corrected using a predetermined deviation table. The main impediments to navigational accuracy are the heading reference and the speed over ground measurement. In this system, the heading reference is derived from both the Honeywell compass and the Systron Donner IMU, which provides yaw rate. The fusion of the yaw rate and the compass data leads to an identification of the yaw rate bias, which is modeled by a constant value. The compass bias which is mostly dependent on the vehicle heading relative to magnetic north, is automatically identified in the EKF, [9] and [11] using DGPS positions when surfaced. When submerged, the position error covariance grows, but is corrected on surfacing. By the nature of the EKF any new GPS update corrects the estimation of all states. Since all states are locally observable by the GPS channels, only a relatively short surface time (related to the filter response time, for example, 10 seconds) allows the filter to re-estimate biases, correct position estimates and continue with improved accuracy. Underwater, the heading bias is unobservable and will remain fixed until a new update is obtained.

As a demonstration, the ARIES vehicle was operated in Monterey Bay in a series of runs including a dive-surface-dive-surface sequence. Figure 6, below shows a plot of vehicle position in an exercise where the vehicle is commanded to follow a track at depth, come up for a DGPS correction, then follow the bottom at an altitude of 3m, while a video is recorded from a down-looking camera. The vehicle then surfaces to get a second fix before turning round and repeating the exercise from the complementary heading. In this plot, the vehicle trajectory is designed to fly over the Monterey Inner Shelf Observatory (MISO) instrument frame placed in 12 meters of water approximately 0.5 kilometers from shore with estimated GPS position used to design the approach lane. The video taken as the vehicle flies over the MISO is designed to provide identification details of the object given its approximate DGPS location (denoted by the large asterisk in Figure 6a).

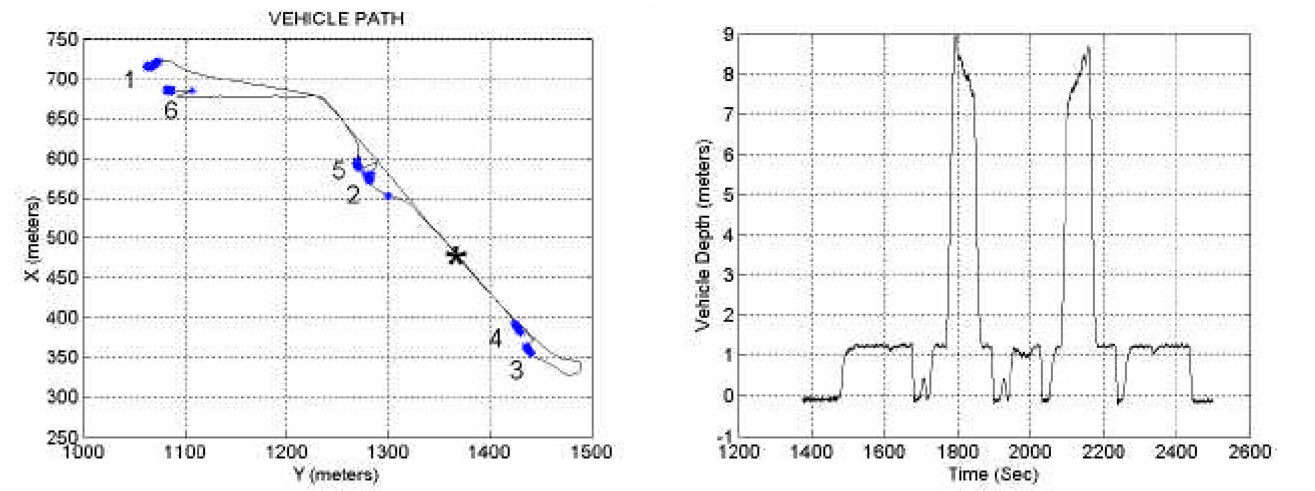

Figure 6a. Vehicle path showing locations where the GPS position fixes were obtained by surfacing for 20 seconds (asterisks). 6b. Depth response during run that clearly shows the DGPS pop up maneuvers.

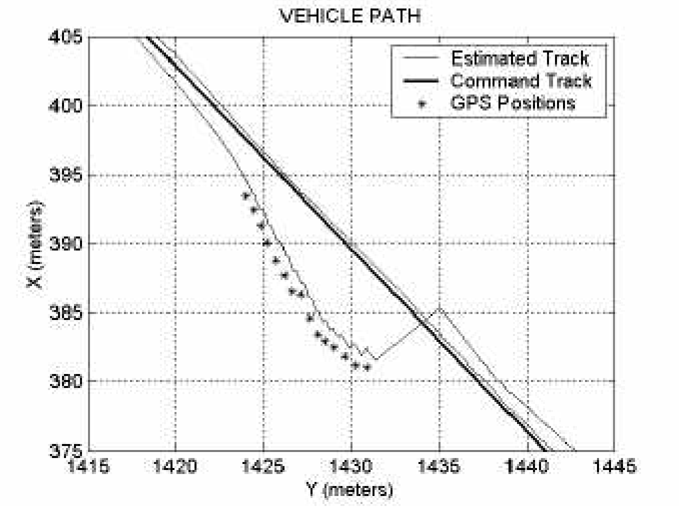

Figure 7. Close up of the final surface showing the filter solution together with the DGPS measurement at the surface (position 4 in Figure 6a).

In Figure 7, a close up of the final surfacing maneuver shows that there is great consistency in estimating the true DGPS data point as seen by the AshTech G-12 unit on board. The difference between the Kalman Filter solution and the DGPS data points while surfaced is sub meter precision. However, the difference between the dead reckoning solution underwater is a few meters off the mark.

In Figure 8, the number of visible satellite vehicles seen by the GPS unit are shown to evolve quickly. Within 10 seconds, 8 satellites are being used to compute the position solution, with a corresponding reduction in the horizontal dilution of precision (Hdop). Values of Hdop between 1.2 and 1.7 are usually associated with high precision but are just figures of merit used in the G-12 sensor and not related directly to error covariances (see [6] for more details). A value of 1 is shown when differential GPS is available and 0 otherwise.

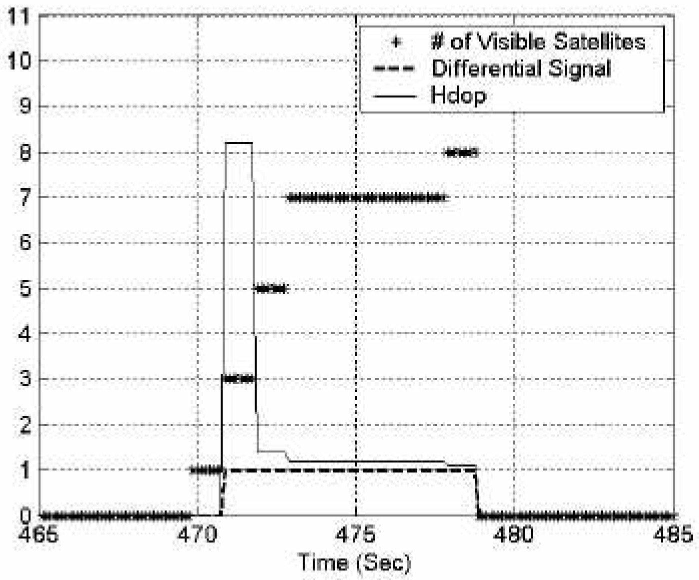

Figure 8. Time history of the response of the number of visible GPS satellites, presence of the differential signal, and Hdop during a routine surface popup. Note: This particular does correspond to that in Figure 7 but is provided to show the Hdop and the differential signal response.

Figure 9 shows the response of the heading bias estimate from the EKF for the entire run. At each surface approximately 10 DGPS points are obtained which rapidly corrects the compass bias. However, as is seen, compass corrections in the neighborhood of 5 degrees are still needed to predict correctly the vehicle positions.

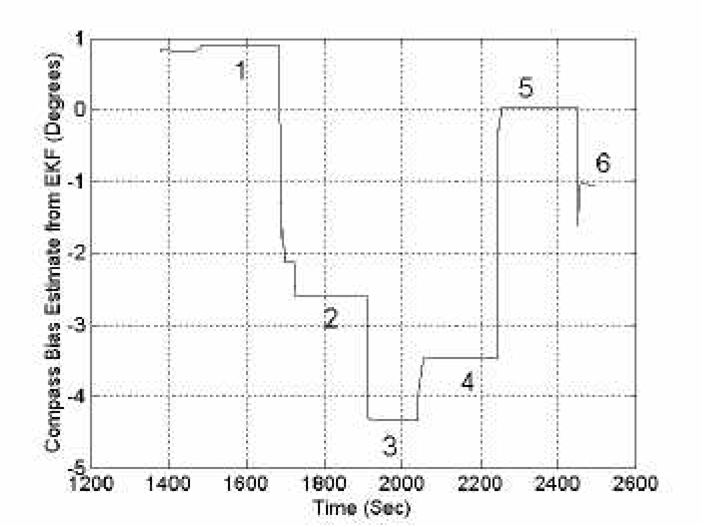

Figure 9. Compass bias estimate versus time. Positions 1-6 are correlated with figure 6a.

This is an indication that further corrections of the compass deviation table are needed. The remaining question is whether or not the deviations are predictable or random. While some additional runs suggest that there may be some degree of consistency, it remains to be shown conclusively.

In spite of the above, the navigation accuracy was sufficient to identify the MISO Laboratory 4 times out of 5 passes using video. The photo in Figure 10 shows an image of the structure from 3 m above bottom.

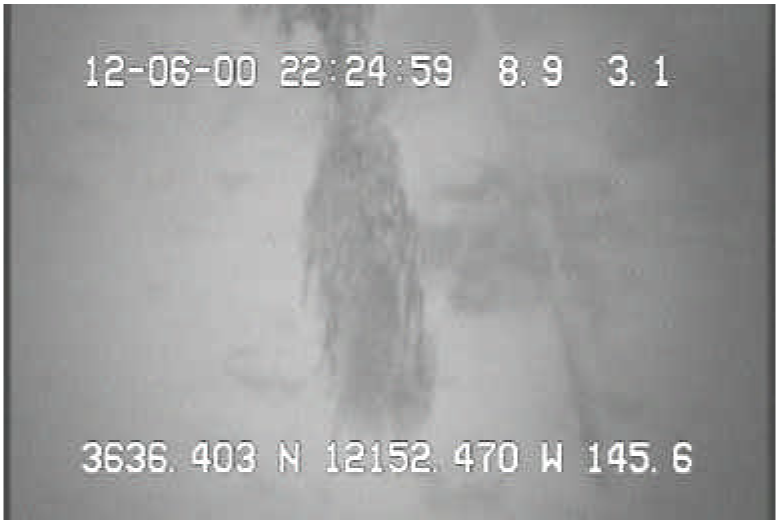

Figure 10 Photo of MISO frame structure from 3 m above bottom in 12 m water depth.

VII. CONCLUSIONS

This paper has described a third generation AUV from the NPS Center for AUV Research. A new computer architecture has been described to enable the vehicle to operate as a network server using acoustic and radio communication links. The vehicle is capable of automatically controlling its altitude above bottom, its depth, its heading, and cross-track error in way point based missions. A new dual processor computer architecture has been described and tested with highly reliable results. The architecture allows for the integration of concurrent data from all sensors, including inputs from sonars, a C^2 radio link and an underwater acoustic link. Inter-processor networking is provided. Data files have been transferred from the ARIES to a command station at 2000-3000 bytes per second over ranges up to 6 km. in the present configuration. The low data rate acoustic link has been tested successfully. Most importantly, the vehicle has been designed for accurate navigation in shallow water using an extended Kalman filter and DGPS. In spite of many efforts to generate an accurate compass deviation table with less than one degree of error, heading errors larger than this, appear to remain. Position errors less than 5 meters can be found using a second DGPS correction in most cases. The tracking control has been proven to be sufficient to obtain video capture of a small underwater structure given its known position.

To obtain position errors within 1 meter, extreme care in developing the compass deviation table must be taken. In the future to get errors within 1% of distance traveled will require a high quality, high cost, navigation grade IMU. The results have shown that the control errors are very small and all controllers are well behaved and 8 to 9 GPS satellites can be acquired with 10 seconds of surfacing.

APPENDIX A

TABLE A1 Example `Track.out` file for way point navigation.

Column # ®

1	2	3	4	5	6	7	8	9	10	11
596.66	1286.09	700.0	700.0	0	1.75	2.0	1	25.0	8.00	40.0
533.43	1326.44	700.0	700.0	0	1.75	2.0	1	25.0	8.00	150.0
471.02	1368.04	500.0	500.0	1	1.75	2.0	0	25.0	8.00	100.0
408.61	1409.64	500.0	500.0	1	1.75	2.0	0	25.0	8.00	100.0
330.40	1463.58	700.0	700.0	0	1.75	2.0	1	25.0	8.00	150.0
409.16	1410.47	700.0	700.0	0	1.75	2.0	1	25.0	8.00	200.0
471.57	1368.87	500.0	500.0	1	1.75	2.0	0	25.0	8.00	100.0
533.98	1327.27	500.0	500.0	1	1.75	2.0	0	25.0	8.00	100.0
596.66	1286.09	700.0	700.0	0	1.75	2.0	1	25.0	8.00	150.0
534.54	1328.11	700.0	700.0	0	1.75	2.0	1	25.0	8.00	200.0
472.13	1369.71	500.0	500.0	1	1.75	2.0	0	25.0	8.00	100.0
409.72	1411.31	500.0	500.0	1	1.75	2.0	0	25.0	8.00	100.0
330.40	1463.58	700.0	700.0	0	1.75	2.0	1	25.0	8.00	150.0
410.27	1412.14	700.0	700.0	0	1.75	2.0	1	25.0	8.00	200.0
472.68	1370.54	500.0	500.0	1	1.75	2.0	0	25.0	8.00	100.0
535.09	1328.94	500.0	500.0	1	1.75	2.0	0	25.0	8.00	100.0
596.66	1286.09	700.0	700.0	0	1.75	2.0	1	25.0	8.00	150.0
646.58	1252.81	700.0	700.0	0	1.75	2.0	0	25.0	8.00	100.0

Each row describes attributes for a single way point. Columns 1-11 are defined by:

Col. # Description

1 X position way point (meters).

2 Y position way point (meters).

3 Left screw command speed (rpm).

4 Right screw command speed (rpm).

5 Control mode flag, 0 = Depth Control, 1 = Altitude Control.

6 Commanded Altitude (meters, if applicable).

7 Commanded Depth (meters, if applicable).

8 Perform GPS popup on this track, 1 = Yes, 0 = No.

9 Duration of GPS popup (Sec).

10 Watch Radius, $R_{w(i)}$ (meters).

11 Way point timeout (Sec). If the current way point is not reached within this time, the mission is aborted.

ACKNOWLEDGEMENTS

The authors wish to thank the Office of Naval Research (Dr. Tom Curtin and Dr. Tom Swean) for the financial support of this project.

REFERENCES

[1] T. Curtin, J. Bellingham, J. Catapovic, and D. Webb, “Autonomous Oceanographic Sampling Network” *Oceanography*, vol. 6, no. 3, pp. 86-94,1993.

[2] S. M. Smith, K. Ganesean, P. E. An, and S. Dunn, "Strategies for Simultaneous Multiple AUV Operation and Control", *International Journal of System Science*, vol. 29, no. 10, pp. 1045-1063, 1998.

[3] B. Allen, R. Stokey, T. Austin, et. al., “REMUS: A Small Low Cost AUV: System Description, Field Trials, Performance Results”, *Proceedings IEEE Oceans 97*, pp. 994-1000, 1997.

[4] AUVFest ’99, NAVO Report http://www.cnmoc.navy.mil/ auvdemo/index.htm.

[5] D. B. Marco, “Autonomous Control of Underwater Vehicles and Local Area Maneuvering”, Ph.D. Dissertation, Naval Postgraduate School, 1996.

[6] Ashtech Products,G12 Sensor, http://ashtech.com/Pages/prodoem.htm.

[7] L. R. LeBlanc, M. Singer, P. Beaujean, et. al., “Improved Chirp FSK Modem for High Reliability Communications in Shallow Water”, *Proceedings IEEE Oceans 2000*, IEEE #00CH37158C,pp. 601-603, 2000.

[8] A. J. Healey, and D. Lienard, “Multivariable Sliding Mode Control for Autonomous Diving and Steering of Unmanned Underwater Vehicles”*, IEEE Journal of Oceanic Engineering*, Vol. 114, No.3, July, 1993.

[9] P. E. An, A. J. Healey, J. Park, and S. M. Smith, "Asynchronous Data Fusion For AUV Navigation Via Heuristic Fuzzy Filtering Techniques", *Proceedings IEEE, Oceans 97*, Halifax, Oct. 1997 IEEE CD-ROM 0-7803-4111-2 http://web.nps.navy.mil/~me/healey/papers/oceans_97.pdf

[10] Y. Kanayama, and B. I. Hartmann, "Smooth Local Path Planning for Autonomous Vehicles" *Autonomous Robot Vehicles*, Springer-Verlag, ISBN 000-387-97240-4, pp. 62-68, 1990.

[11] A. J. Healey, P. E. An, and D. B. Marco, "On Line Compensation of Heading Sensor Bias for Low Cost AUV's", *Proceedings of the IEEE Workshop on Autonomous Underwater Vehicles*, AUV '98, IEEE Catalog. Number 98CH36290, ISBN # 0-7803-5190-8, August 20-21, 1998, Cambridge, Mass. pp 35-42 http://web.nps.navy.mil/~me/healey/papers/auv98.pdf